Differential Transform for Video-based Plenoptic Point Cloud Coding

Diogo C. Garcia, *Senior Member, IEEE,* Camilo Dorea, Renan U. B. Ferreira, Davi R. Freitas, Ricardo L. de Queiroz, *Fellow, IEEE,* Rogerio Higa, Ismael Seidel, Vanessa Testoni

***Abstract*—Point cloud compression has been studied in standard bodies and we are here concerned with the Moving Picture Experts Group video-based point cloud compression (V-PCC) solution. Plenoptic point clouds (PPC) is a novel volumetric data representation wherein points are associated with colors in all viewing directions to improve realism. It is sampled as a number (N_c) of attribute colors per point. We propose a new method for the efficient video-based compression of PPC that is backwards compatible with the existing single-color V-PCC decoder. V-PCC generates three image atlases which are encoded using an image/video encoder. We assume there may be a reference color which is to be encoded as the main payload. We generate $N_c + 3$ atlases and we produce N_c differential images against the reference color image. Those difference images are pixel-wise transformed using an N_c-point discrete cosine transform, generating N_c transformed atlases which are encoded, forming the secondary payload. Such secondary information is the plenoptic enhancement to the point cloud. If there is no reference attribute, we skip the differences and use the lowest frequency of the transformed atlases as the main payload. Results are presented that show an unrivaled performance of the proposed method.**

***Index Terms*—Point-cloud compression, video-based point cloud compression.**

I. Introduction

POINT clouds have recently gained increased attention in the representation of volumetric information [1]–[6]. Their simplicity compared to meshes simplifies the real-time capture and rendering of a point cloud (PC). However, the amount of data necessary to represent volumes with point clouds is very large, which demands efficient compression techniques [7]–[9] . Among many groups and organizations interested in compressing PCs, the Moving Picture Expert Group (MPEG) has advanced its own point cloud compression (PCC) standards. We are here interested in MPEG's video-based point cloud compression (V-PCC) standard for the encoding of voxelized point clouds, *i.e.* clouds where the points are quantized into volumetric elements (voxels) [10]–[12].

A novel volumetric data representation is the plenoptic point cloud (PPC), which offers greater rendering realism by presenting different colors according to the viewing direction [13]. The name stems from the 7D plenoptic function

$$p(x, y, z, \theta, \phi, \lambda, t), \quad (1)$$

which represents the component at any point (x, y, z) from any direction (θ, ϕ) with a given wavelength (λ) at any given time (t) [14]. Our approach to represent such a function begins

D. Garcia and R. Ferreira are with the Gama Engineering College, University of Brasilia, Brazil, e-mail: renan,diogogarcia@unb.br

R. de Queiroz and C. Dorea are with the Department of Computer Science, University of Brasilia, Brazil, e-mail: queiroz@ieee.org and camilodorea@unb.br

D. Freitas is with the Electrical Engineering Department, University of Brasilia, e-mail: rabbouni.davi@image.unb.br

R. Higa, I. Seidel and V. Testoni are with Samsung R&D Institute Brazil, Campinas, Brazil, e-mail: r.higa,i.seidel,vanessa.t@samsung.com

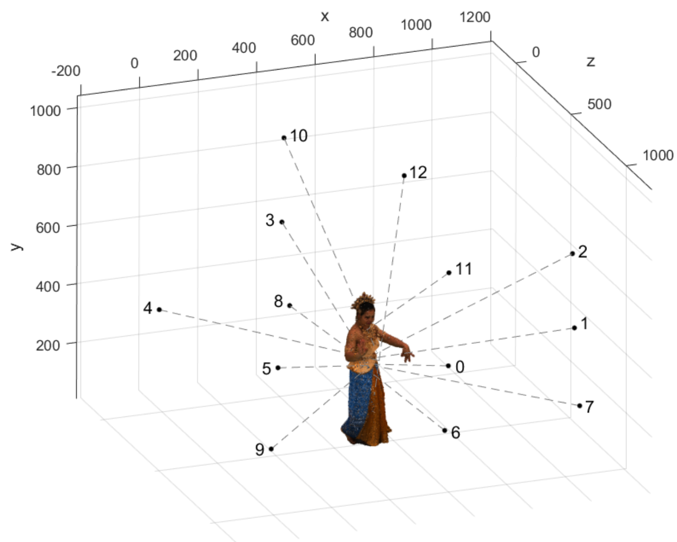

Figure 1. Camera rig positions, labeled 0 through 12, around subject for the plenoptic point cloud *Thaidancer*.

Figure 2. Rendering of the plenoptic point cloud *Thaidancer* as seen from cameras 5, 6 and 7. Note how specularity can cause the same voxel to have different apparent colors at different viewing angles, specially on the golden fabric of the model.

with its discretization within a point cloud representation. The spatial coordinates (x, y, z) are discretized into voxel geometry (position) information. The wavelength λ is discretized into the red-green-blue (RGB) colors and t is discretized when we capture dynamic point cloud frames. The (θ, ϕ) pair represents the polar and azimuth angles of a sphere centered at (x, y, z), essentially the viewing direction to a voxel. We sample p at a number of (θ, ϕ) pairs and assume the renderer is to interpolate the function at in-between directions. Furthermore, since the point cloud is captured with a finite number of camera rigs, it is natural to sample (θ, ϕ) at the position of the cameras relative to each voxel, to avoid extra data interpolation, as illustrated in Fig. 1 for the plenoptic image *Thaidancer*. Therefore, the plenoptic function is well represented as a dynamic point cloud with multiple color attributes per voxel, one for each of the N_c cameras (camera rigs). It is represented by geometry (xyz integer triplets) and multiple color information $\{rgb_n\}$, $0 \leq n < N_c$, and we refer to it as a PPC [13]. To illustrate the richness of detail gained from the plenoptic representation, Fig. 2 presents the rendering of *Thaidancer* as seen from cameras 5, 6 and 7: note the changes in the lighting of the model's golden fabric, specially at her waist. PPCs typically come with a default RGB value triplet (rgb) per voxel, which we take advantage for backward compatibility with V-PCC.

PPCs require much more data for representation than regular PCs, and they can greatly benefit from PC compression techniques, which are an actively-studied subject [10][11]. Sandri *et al.* have proposed the compression of PPC by many forms, of which the use of the Karhunen-Loève transform (KLT) over voxel colors and region-adaptive hierarchical transform (RAHT) over the transformed PCs has demonstrated to be the most effective [13]. Zhang *et al.* have a different view of the PPC representation, where the color function over (θ, ϕ) is continuous instead of sampled. The function is parameterized and parameters are encoded and transmitted to the decoder [15]. Naik *et al.* proposed the incorporation of multiple color attributes into V-PCC test model [16]. They later proposed discarding some views and using the remaining ones to interpolate the former [17],[18]. Li *et al.* proposed a video-based solution based on V-PCC adapted for multiple attributes. However, instead of encoding each transformed atlas with a single High Efficiency Video Coder (HEVC), they use the multiview HEVC (MV-HEVC) [19].

Section II describes our method as it relates to V-PCC and Sec. III explain the transform and encoding methods. Section IV has the description and results of our simulations and Sec. V contains the conclusions of this work.

II. Plenoptic enhancement for V-PCC

MPEG's V-PCC standard currently associates one attribute, such as color, per voxel. The plenoptic information can be represented as extra color attributes, one for each camera, which are compressed using a standard video codec [20]. Our method uses this framework as a base and extends it through a differential transform applied to the atlases rendered for each color attribute. We want to support PPC within V-PCC with a backwards-compatible framework, *i.e.*, our compressed stream could be decompressed by a *stock* V-PCC decoder generating a regular single-color-attribute point cloud.

V-PCC projects different patches around the 3D point cloud onto 2D images, forming mosaics called atlases. The main encoded information is actually composed of 3 atlases, namely depth, occupancy and attributes, along with some support information. The depth and occupancy atlases are related to the geometry information, while the attribute atlas may represent color (texture), material or transparency of the point cloud, among others [11].

Since this paper deals with the PC's plenoptic color information, we only refer to its texture from here on. The PC's texture colors are conveyed as image pixel colors, in the atlases, and not all voxels are mapped to pixels. Hence, we only process the atlas pixels, instead of the point cloud attributes. This approach is more practical, avoids wasteful computation, and allows us to process locally-decoded reference images, in order to improve accuracy.

The atlases are images that have already been converted from RGB to a YUV color space, so that the chrominance (U and V) components can be spatially decimated in order to improve compression at lower rates. The conversion from RGB to YUV brings another advantage from the fact that camera color variation is most prominent in the luminance component, which best reflects changes in specularity [18]. Figure 3 shows measures of the variance across the N_c camera colors for the different components and for different point clouds. We can see that chrominance variance is quite small regardless of the point cloud. This is a specularity property of the surfaces involved that can be explored to enhance compression.

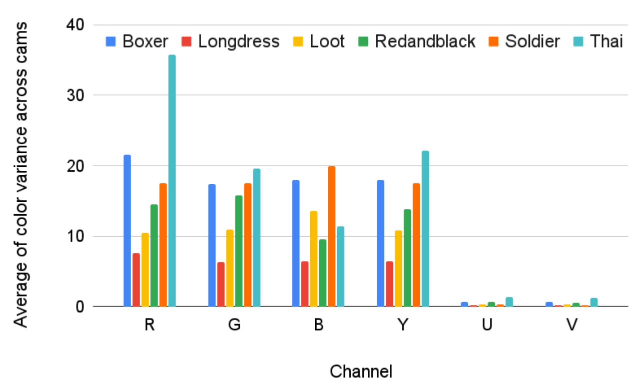

Figure 3. Color variance across plenoptic cameras, averaged over all voxels, for RGB and YUV color components.

In order to accommodate PPCs into V-PCC, we propose an extension that is compatible with existing decoders. For this, the encoder generates two payloads that are streamed to the decoder: the main and the plenoptic enhancement payloads. The main payload is regular V-PCC encoded data that is transparently decoded by a non-plenoptic (unaware) decoder. The main payload carries the information of a single color per voxel and the decoder reconstructs a regular point cloud. The plenoptic enhancement payload is supplemental and carries the information for all the N_c colors per voxel. An "aware" decoder would use the main payload to reconstruct a single-color PC and, then, to add the information of all the color attributes.

We refer to such a coder as the video-based plenoptic point cloud compressor (V-PPCC). Figure 4 depicts diagrams of the V-PPCC encoder and decoder.

The important part to note is the presence or not of a main color along with additional plenoptic colors, depending on the application. Many PPCs come with a main RGB channel, which could be one of the N_c cameras, a weighted average of them [22], or any other combination of interest. It is assumed to be conveyed to the decoder and is used as the main payload. Then, the plenoptic enhancement payload carries the information of all the N_c other colors. If the main RGB channel is not present, the proposed encoder may create it from the N_c camera colors, or it can simply use the lowest frequency (DC) atlas as the main payload, while the plenoptic enhancement carries information about the other $N_c - 1$ channels which would be necessary to reconstruct the N_c colors per voxel. The latter option avoids the creation of yet another texture atlas at the encoder side.

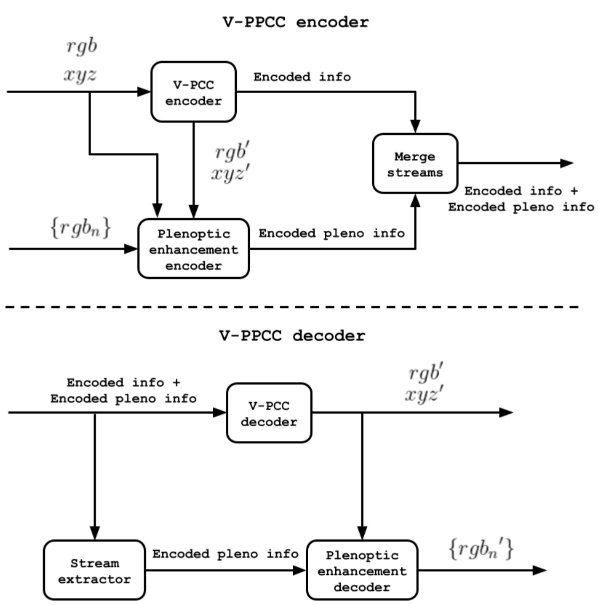

Figure 4. Proposed backward-compatible plenoptic V-PCC codec (V-PPCC).

III. Transform coding

For backwards compatibility, we consider two scenarios depending on whether the main RGB channel is conveyed to the encoder or not. If so, the main payload is the main RGB channel as a texture atlas, sent along with the geometry-related information. In this case, the plenoptic enhancement contains N_c texture atlases and is discussed in Sec. III-A. In the scenario where the main RGB channel is not present, the main payload is one of the N_c texture atlases we generate, which is sent along with the geometry information. The plenoptic enhancement is, then, made of $N_c - 1$ texture atlases and the process is described in Sec. III-B.

A. Plenoptic differential coding

For the differential encoder depicted in Fig. 5, the following is proposed:

- In order to maintain compatibility with standard V-PCC, the differential encoder compresses the geometry and the main color attribute information through a standard V-PCC. This process initially generates an occupancy atlas $Occ(i, j)$ and a geometry atlas $Geom(i, j)$, where i and j are pixel coordinates.
- These two atlases are used to map the main color attributes into the corresponding 2D texture atlas, which we refer as $YUV(i, j)$, after conversion from RGB to YUV color space.
- All this information is encoded into the main payload, and a local reconstruction $YUV'(i, j)$ of $YUV(i, j)$ is generated.
- The same process of texture atlas formation is applied to each of the N_c color attributes, generating a set of $\{YUV_n(i, j)\}$ images. The encoder then constructs a set of differential signals $\{E_n(i, j)\}$, where $E_n(i, j) = YUV_n(i, j) - YUV'(i, j)$.
- Each YUV color channel of the differential signal set is considered separately and is subjected to a discrete cosine transform (DCT) across cameras, per pixel position, generating N_c images of coefficients $\{DCT_n(i, j)\}$. Note that $DCT_n(i, j)$ also is an YUV image itself.
- All coefficient images are then uniformly scaled to fit an integer M-bit representation, for example, 8-bits in HEVC, by adding a constant value 2^{M-1} to all coefficients (low and high frequencies) and rounding them to the nearest integer. The scaled coefficient images are encoded as video, through HEVC, and the corresponding streams are merged.

The reverse process is applied at the decoder, as depicted in Fig. 6, *i.e.*, video streams are separated and decoded. The streams corresponding to the additional plenoptic color attributes are de-scaled through subtraction, inverse transformed and added to the decoded version of $YUV(i, j)$.

B. Plenoptic non-differential coding

The non-differential coding process is outlined in Fig. 7, according to the following algorithm:

- When encoding PPC content which does not contain a main color attribute set, the DC coefficient image resulting from the DCT is adopted in place of the main color attribute.
- Unlike the differential encoder of Sec.III-A, the DC coefficient image is multiplied by $1/\sqrt{N_c}$ and rounded to the nearest integer. Remaining AC coefficient images are not scaled but offset by a constant value 2^{M-1} to all coefficients, before rounding them to the nearest integer.
- All scaled coefficient images are, then, HEVC-encoded.
- Effectively, the reference RGB channel is set to zero and the main payload texture is replaced with the DC texture atlas.

Non-differential decoding proceeds in the inverse fashion as illustrated in Fig. 8. After stream separation, the DC coefficient

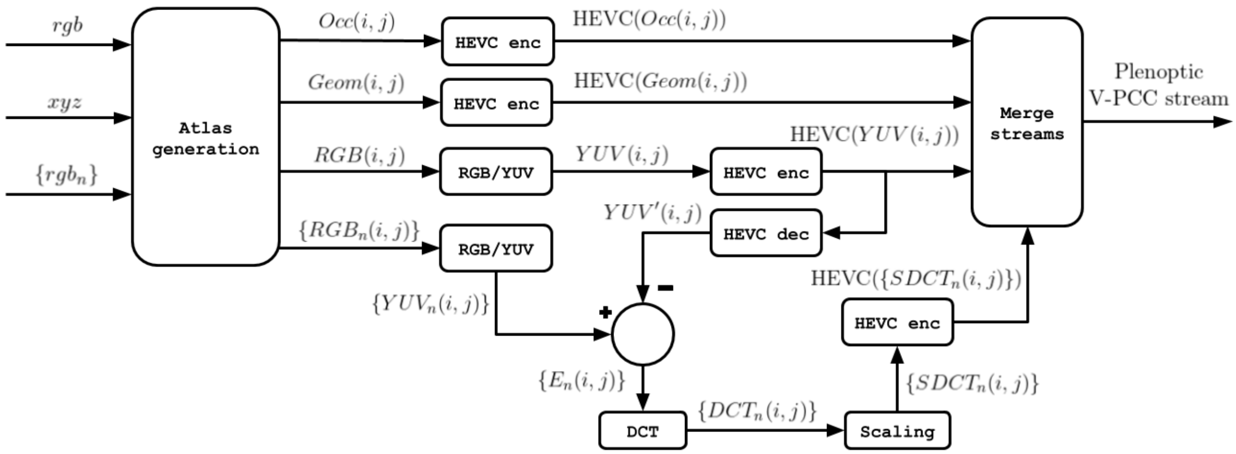

Figure 5. Plenoptic enhancement differential encoder.

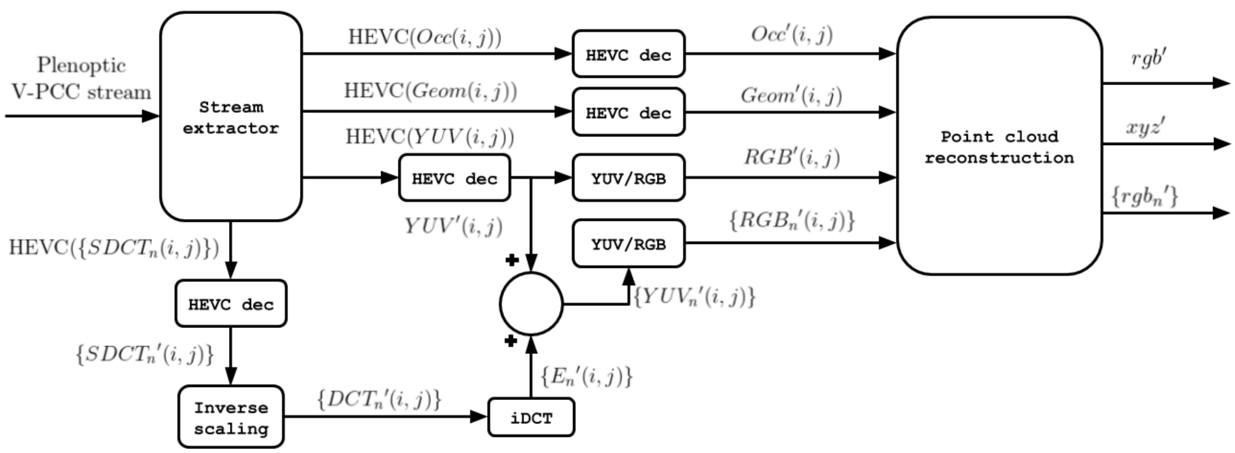

Figure 6. Plenoptic enhancement differential decoder.

image is HEVC-decoded along with the occupancy, geometry and AC-coefficient images. The DC and AC coefficients are reintegrated, de-scaled, and the inverse DCT is performed in order to produce the decoded plenoptic color attributes.

IV. Compression experiments

The proposed codecs were implemented in version 9.0 of the V-PCC reference software Test Model Category 2 (TMC2) [20][21] and compared to the current solution supported by V-PCC, *i.e.*, encoding the plenoptic information as multiple point cloud attributes, and to the work by Li *et al.* [19] (which is video-based but is not V-PCC-compatible). The "Intra" configuration for the V-PCC Common Test Conditions (CTC) [23] was applied in all tests. Separate rate and distortion values were calculated for each of the scenarios proposed in Section III.

Experiments were carried out with the 8i Voxelized Surface Light Field Dataset (8iVSLF) [22]. Each of the PPCs in this set has, for each voxel, a set of RGB color values associated with a different camera viewpoint. N_c is either 12 or 13 and voxel positions lie within a cube of $4096 \times 4096 \times 4096$ voxels, which is known as "depth 12" or "vox12" resolution. Besides plenoptic color attributes, the dataset includes a main RGB channel which either consists of color from the front-facing camera or from a weighted average of the cameras. Table I provides details for each of these PPCs.

Since the TMC2 software does not present the individual rates for the Y, U and V channels, we adopt a combined distortion of the three channels for proper comparisons. We calculate the distortion as a weighted average of the Peak-Signal-to-Noise Ratio (PSNR) for Y, U and V channels:

$$\mathrm{PSNR_{YUV}} = \frac{6\ \mathrm{PSNR_Y} + \mathrm{PSNR_U} + \mathrm{PSNR_V}}{8}. \qquad (2)$$

Rather than averaging PSNRs across all cameras, we calculate PSNR from the $N_c\!+\!1$ cameras as a single signal. We consider this interpretation more intuitive since PSNR averaging is, in essence, a geometric mean of the Mean Square Error (MSE) of each camera.

A. Plenoptic differential coding

The scenario proposed in Subsection III-A considers that a main RGB channel is available along with N_c cameras,

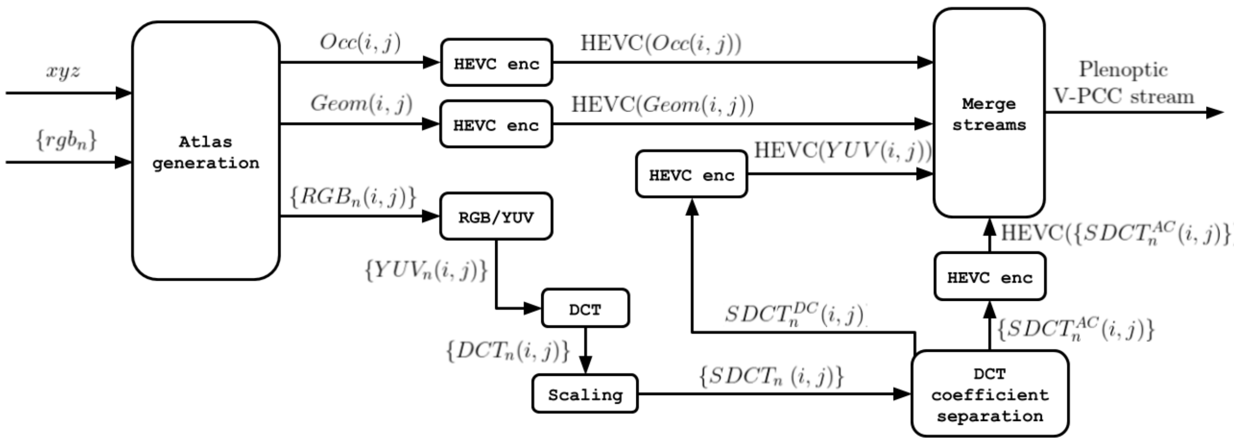

Figure 7. Plenoptic enhancement non-differential encoder.

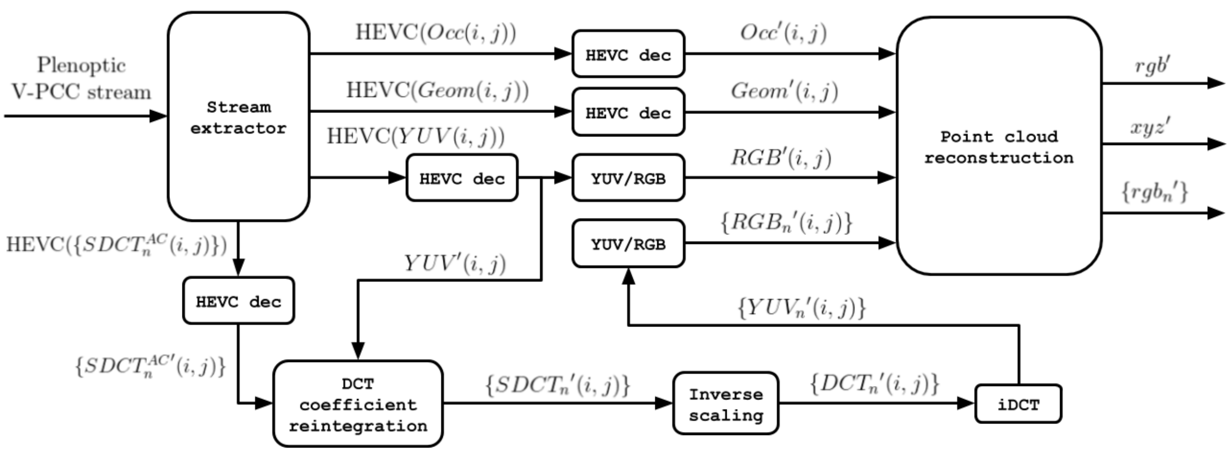

Figure 8. Plenoptic enhancement non-differential decoder.

Table I
8IVSLF DATASET.

Dataset	# voxels	N_c
Boxer	3493085	13
Longdress	3096122	12
Loot	3017285	13
Redandblack	2770567	12
Soldier	4001754	13
Thaidancer	3197804	13

Table II
WEIGHTED YUV-PSNR BD-RATE FOR PLENOPTIC DIFFERENTIAL CODING OVER V-PCC.

Dataset	BD-Rate
Boxer	-72.8%
Longdress	-86.7%
Loot	-67.2%
Redandblack	-81.1%
Soldier	-74.7%
Thaidancer	-82.4%

so that its rate and distortion values should be considered in all evaluated methods for a proper comparison of coding performances. In that manner, we calculate the rate as the total number of attribute bits considering the $N_c + 1$ cameras, and the distortion as the weighted PSNR (2) of the N_c+1 cameras.

Table II presents the Bjøntegaard Delta rate (BD-Rate) for plenoptic differential coding over V-PCC. Clearly, the proposed method offers substantial gains over the default solution, encoding the plenoptic information as multiple point cloud attributes. Fig. 9 presents the rate-distortion performance of both methods for *Thaidancer*. The results for the other PPCs share a similar pattern.

In the work by Li *et al.*, the distortion was calculated as the average PSNR across N_c cameras for the Y channel [19]. We believe our way of calculating the distortion is fairer, for the reasons explained above. Nonetheless, we present corresponding results for the proposed method, for adequate comparisons. We included the rate and the distortion from the main RGB channel to Li *et al.*'s results, accordingly.

Table III presents the BD-Rate for plenoptic differential coding and Li *et al.*'s method over V-PCC. The proposed method offers greater rate reduction on average for all PPCs, except for *Thaidancer*, where the difference is of less than

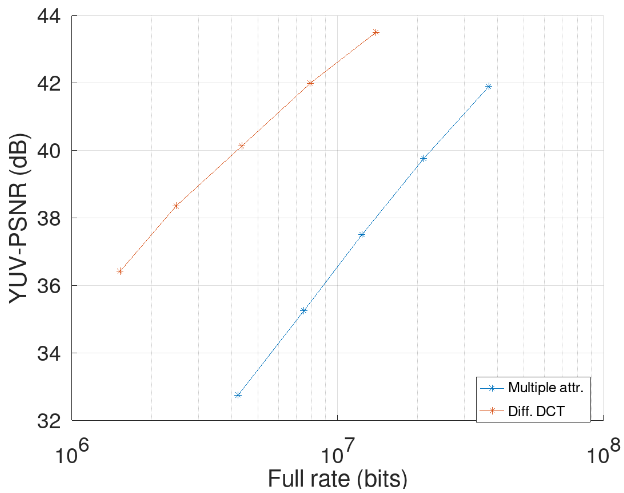

Figure 9. Rate-distortion performance for *Thaidancer* for the plenoptic enhancement differential codec ("Diff. DCT") and V-PCC ("Multiple attr.").

1%. Fig. 10 presents the rate-distortion performance for the three methods for *Thaidancer*, which is representative of all PPCs. Both methods encode transformed prediction residuals. However, our method transforms data among cameras as well, yielding better results at smaller rates. Since the DCT coefficients in our method are rounded prior to being encoded by HEVC, this limits the maximum quality attainable. Hence, both methods have similar performance at higher rates.

Table III
Y-PSNR BD-RATE FOR LI *et al.* AND PLENOPTIC DIFFERENTIAL CODING OVER V-PCC.

Dataset	Li *et al.*	Plenoptic diff. coding
Boxer	-57.1%	-74.5%
Longdress	-81.3%	-85.5%
Loot	-61.5%	-67.9%
Redandblack	-71.3%	-80.0%
Soldier	-68.7%	-74.7%
Thaidancer	-79.8%	-80.9%

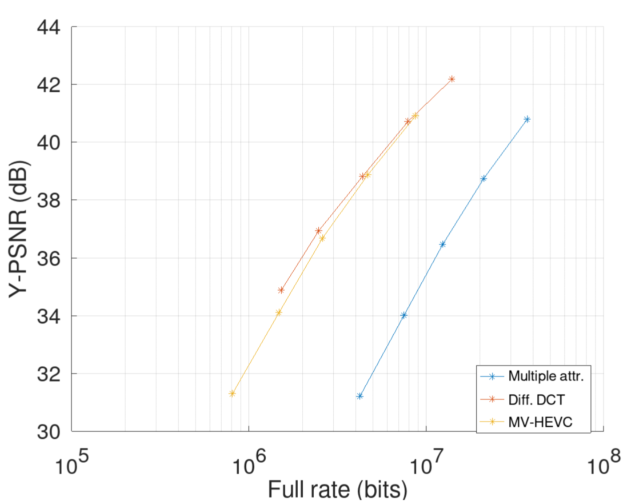

Figure 10. Rate-distortion performance for *Thaidancer* for the plenoptic enhancement differential codec ("Diff. DCT"), V-PCC ("Multiple attr.") and the method of Li *et.al.* ("MV-HEVC").

We also compared the performance of the proposed method applying the KLT instead of the DCT. Figure 11 shows the results for *Longdress*, where it can be seen that the KLT offers little rate-distortion gain over the DCT, at a much higher complexity cost.

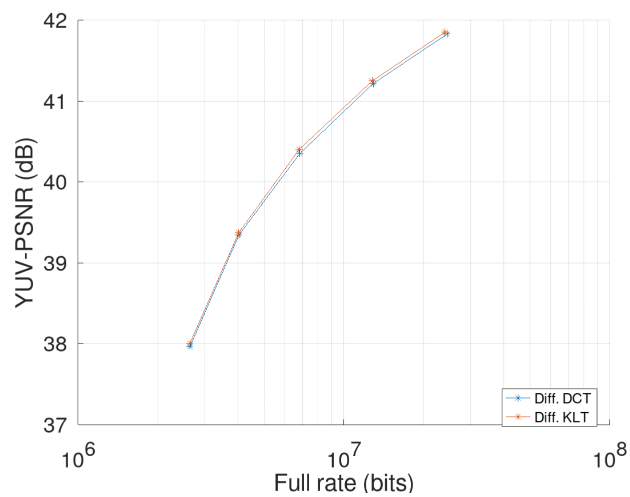

Figure 11. Rate-distortion performance for *Longdress* for the plenoptic enhancement differential codec using the DCT ("Diff. DCT") and the KLT ("Diff. KLT").

B. Plenoptic non-differential coding

The scenario proposed in Subsection III-B considers that the main RGB channel is not available, only the N_c plenoptic cameras are present. Thus we disregard any main RGB channel attributes of the dataset in coding and associated rate and distortion values.

Table IV presents the BD-Rate for plenoptic non-differential coding over V-PCC. Once again, the proposed method offers substantial gains over the default solution. Fig. 12 presents the rate-distortion (RD) performance of both methods for *Thaidancer*. The results for the other PPCs share a similar pattern.

Table IV
WEIGHTED YUV-PSNR BD-RATE FOR PLENOPTIC NON-DIFFERENTIAL CODING OVER V-PCC.

Dataset	BD-Rate
Boxer	-77.4%
Longdress	-88.9%
Loot	-75.4%
Redandblack	-84.0%
Soldier	-81.1%
Thaidancer	-86.6%

In order to compare the proposed method with Li *et al.*'s codec, we present on Table V corresponding results for the BD-Rate using the average PSNR across N_c cameras for the Y channel. Li *et al.*'s results are different from those in Table III because the rate and the distortion from the main RGB channel were not included in Table V. From Table V, we can see that the proposed method offers greater rate reduction for all PCs. Fig. 13 presents RD curves for the three methods for *Thaidancer*, which is representative of our test set. Once again,

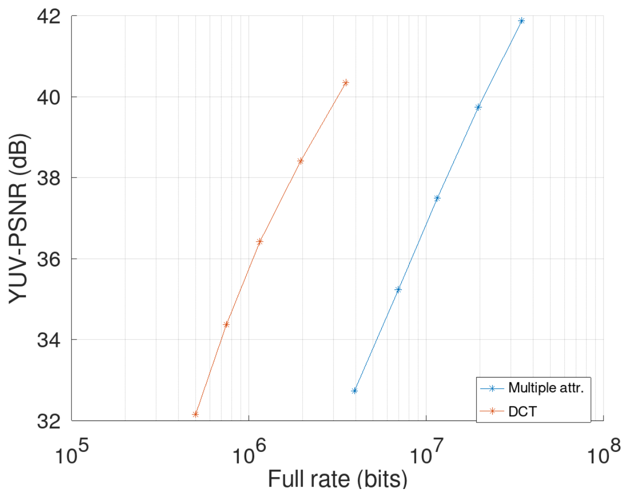

Figure 12. Rate-distortion performance for *Thaidancer* for the plenoptic enhancement non-differential codec ("DCT") and V-PCC ("Multiple attr.").

the proposed method outperforms the others for lower rates. However, the rounding of DCT coefficients prior to HEVC encoding limits the maximum attainable quality. Hence, our method and Li *et al.*'s have similar performance at higher rates.

Table V
Y-PSNR BD-RATE FOR LI *et al.* AND PLENOPTIC NON-DIFFERENTIAL CODING OVER V-PCC.

Dataset	Li *et al.*	Plenoptic non-diff. coding
Boxer	-61.8%	-78.7%
Longdress	-87.8%	-88.7%
Loot	-66.7%	-76.2%
Redandblack	-77.6%	-84.0%
Soldier	-74.2%	-81.2%
Thaidancer	-85.6%	-86.3%

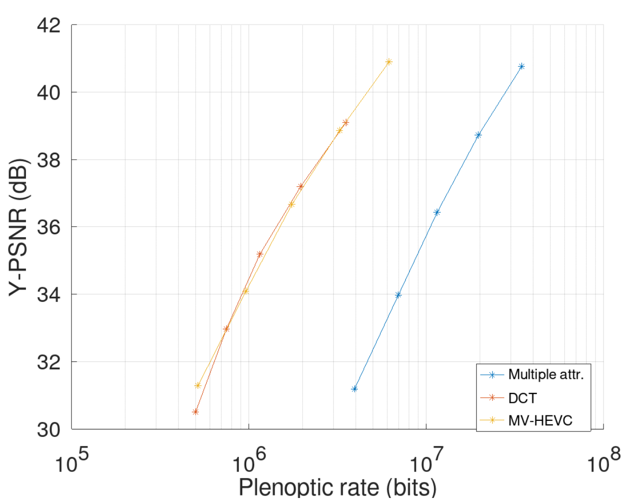

Figure 13. Rate-distortion performance for *Thaidancer* for the plenoptic enhancement non-differential codec ("DCT"), V-PCC ("Multiple attr.") and the method of Li *et.al.* ("MV-HEVC").

Figure 15 presents three views (cameras 5, 6 and 7) of the decoded versions of the PPC *Thaidancer*, after compression with V-PCC, plenoptic differential coding and plenoptic non-differential coding, respectively. All three methods offer similar subjective results, but at a much smaller rate when using the proposed methods.

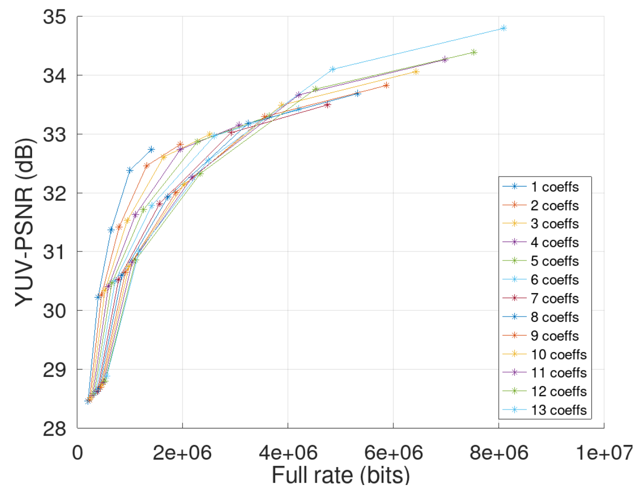

Figure 14. Rate-distortion performance for 10-bit version of *Thaidancer* by applying the proposed method and progressively dropping coefficients.

C. Progressive reconstruction

As a final test, the proposed method was evaluated as DCT coefficients are dropped, from highest to lowest frequency, such that only $N'_C < N_C$ coefficients are transmitted. At the decoder side, the missing coefficients are reconstructed as zero. Figure 14 presents RD curves compressing the 10-bit version of *Thaidancer* with the proposed V-PPCC but progressively dropping coefficients. It can be seen that progressive reconstruction actually outperforms the proposed method for low bitrates, for instance, at 2 Mbits and lower. At higher bitrates (*e.g.* 6 Mbits) fewer coefficients (e.g. one or two) can be dropped if a small YUV-PSNR penalty can be tolerated.

V. CONCLUSIONS

We proposed a novel method for efficient video-based compression of plenoptic point clouds. The method is backward compatible with the existing, single-color, video-based encoder (V-PCC). It can handle plenoptic content containing additional, camera-dependent color attributes. In the first scenario, the differential coder proposal applies a DCT to the difference signals of the atlas images of each of the plenoptic color channels against the main color channel. In the second scenario, our non-differential coder applies the DCT to the plenoptic color-channel atlases and assumes the DC-coefficient image as the main payload, while the AC coefficients make up the secondary payload.

Results, in terms of rate-distortion performance, for the differential and non-differential coders, show substantial gains over the standard coding of plenoptic information (independent coding of the multiple point cloud attributes). Results are also superior to the MV-HEVC solution [19], which is not compatible with V-PCC, but is a competitive video-based codec. Subjective evaluations show that our proposal achieves similar quality to that of standard coding, while operating at significantly lower rates. Further improvements can be made by taking into account inter-frame coding and by optimizing the DCT coefficients' quantization. Nonetheless, we believe the proposed codec is the current state-of-the-art in video-based compression of plenoptic point clouds.

References

[1] J. Lanier, "Virtually There," *Scientific American*, pp. 66–75, Apr. 2001.

[2] C. Zhang, Q. Cai, P. Chou, Z. Zhang, and R. Martin-Brualla, "Viewport: a fully distributed immersive teleconferencing system with infrared dot pattern," *IEEE Multimedia*, vol. 20, no. 1, pp. 17–27, 2013.

[3] J. G. Apostolopoulos, P. A. Chou, B. Culbertson, T. Kalker, M. D. Trott, and S. Wee, "The road to immersive communication," *Proc. IEEE*, vol. 100, no. 4, pp. 974–990, 2014.

[4] A. Collet, et al., "High-quality streamable free-viewpoint video," *ACM Trans. on Graphics*, Vol. 34, No. 4, Article 69, Aug. 2015.

[5] S. Orts-Escolano et al., "Holoportation: virtual 3D teleportation in real-time," *Proc. 29th ACM User Interface Software and Technology Symposium,* Tokyo , Japan, Oct. 2016.

[6] C. Tulvan, R. Mekuria, Z. Li, and S. Laserre, "Use cases for point cloud compression," ISO/IEC JTC1/SC29/WG11 MPEG, document N16331, Jun. 2016.

[7] C. Loop, C. Zhang, and Z. Zhang, "Real-time high-resolution sparse voxelization with application to image-based modeling," in *High Performance Graphics Conf.*, pp. 73–79, 2013.

[8] R. L. de Queiroz and P. A. Chou, "Compression of 3D point clouds using a region-adaptive hierarchical transform," *IEEE Transactions on Image Processing*, vol. 25, no. 8, pp. 3947–3956, Aug. 2016.

[9] R. L. de Queiroz and P. A. Chou, "Motion-compensated compression of dynamic voxelized point clouds," *IEEE Trans. on Image Processing*, Vol. 26, No. 8, pp. 3886–3895, Aug. 2017.

[10] S. Schwarz, M. Preda, V. Baroncini, M. Budagavi, P. Cesar, P. A. Chou, R. A. Cohen, M. Krivokuća, S. Lasserre, Z. Li, J. Llach, K. Mammou, R. Mekuria, O. Nakagami, E. Siahaan, A. Tabatabai, A. Tourapis, and V. Zakharchenko, "Emerging MPEG Standards for Point Cloud Compression," *IEEE J. Emerging Topics in Circuits and Systems,* vol. 9, no. 1, pp. 133–148, March 2019.

[11] D. Graziosi, O. Nakagami, S. Kuma, A. Zaghetto, T. Suzuki and A. Tabatabai, "An overview of ongoing point cloud compression standardization activities: Video-based (V-PCC) and geometry-based (G-PCC)," *APSIPA Transactions on Signal and Information Processing*, 9, E13.

[12] R. Mekuria, C. Tulvan, and Z. Li, Requirements for point cloud compression, ISO/IEC JTC1/SC29/WG11 MPEG, document N16330, Jun. 2016.

[13] G. Sandri, R. L. de Queiroz and P. A. Chou, "Compression of plenoptic point clouds," *IEEE Trans. on Image Processing*, Vol. 28, No. 3, pp. 1419–1427, Mar. 2019.

[14] E. Adelson and J. Bergen, "The plenoptic function and the elements of early vision," *Computational Models of Visual Processing*, pp. 3–20, 1991.

[15] X. Zhang, P. A. Chou, M. Sun, M. Tang, S. Wang, S, Ma and W. Gao, "A Framework for Surface Light Field Compression," *IEEE International Conference on Image Processing*, pp. 2595–2599, Athens, Greece, 2018.

[16] D. Naik and S. Schwarz, "[V-PCC] CE2.15 report on Attribute Coding (SLF)," input document M49123, Gothemburg, Sweden, July 2019.

[17] D. Naik and S. Schwarz, "V-PCC SLF optimisations," in *ISO/IEC JTC1/SC29/WG11 (MPEG/JPEG) document M54620*, July 2020.

[18] D. Naik, S. Schwarz, V. K. M. Vadakita and K. Roimela, "Surface Light-field Support in Video-based Point Cloud Coding," *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, 2020, pp. 1-6.

[19] L. Li, Z. Li, S. Liu and H. Li, "Video-Based Compression for Plenoptic Point Clouds," *Data Compression Conference*, Snowbird, UT, USA, 2020, pp. 378–378.

[20] R. L. de Queiroz, C. Dorea, D. C. Garcia, R. U. Ferreira, D. R. Freitas, R. Higa, I. Seidel and V. Testoni, "Differential plenoptic point cloud coding for V-PCC," in *ISO/IEC JTC1/SC29 Joint WG11/WG7 (MPEG/JPEG) input document M55145*, Online, October 2020.

[21] R. Higa, R. U. Ferreira, D. Freitas, D. C. Garcia, C. Dorea, R. L. de Queiroz, I. Seidel and V. Testoni, "Multi-attribute implementation for multiple colors per point in the TMC2," in *ISO/IEC JTC1/SC29 Joint WG11/WG7 (MPEG/JPEG) input document M55144*, Online, October 2020.

[22] M. Krivokuca, P. A. Chou, and P. Savill, "8i Voxelized Surface Light Field (8iVSLF) Dataset," in *ISO/IEC JTC1/SC29 Joint WG11/WG1 (MPEG/JPEG) input document m42914*, Ljubljana, July 2018.

[23] S. Schwarz, G. Martin-Cocher, D. Flynn, and M. Budagavi, "Common Test Conditions for Point Cloud Compression," Document ISO/IEC JTC1/SC29/WG11 w17766, Ljubljana, Slovenia, Jul. 2018.

Acknowledgment

Part of the results presented in this work was obtained through the "Point Clouds Compression" project, funded by Samsung Eletrônica da Amazônia Ltda., under the Brazilian Informatics Law 8.248/91.

Figure 15. Decoded views (cameras 5, 6 and 7) of the plenoptic *Thaidancer* PC. First row: V-PCC, where the plenoptic data occupies 8488000 bits, or 9193136 bits when considering the default RGB value triplet rgb, in both cases with a PSNR of 34.5 dB with respect to the original point cloud. Second row: plenoptic differential coding, where the plenoptic data occupies 968080 bits when considering rgb, with a PSNR of 34.3 dB. Third row: plenoptic non-differential coding, where the plenoptic data occupies 876160 bits, with a PSNR of 34.1 dB.